Lambton College
In Toronto

Program: ESE 4009_1

INSTRUCTOR: Prof. Mike Aleshams

Group 1

Student Name	Student ID	Signature*
Amonjot Singh Chhina	C0772326	Amonjot Singh Chhina
Anna Joy	C0769402	Anna Joy
Manu Simriti Parbhakar	C0774035	Manu Simriti Parbhakar
Vy Nguyen	C0776242	Vy Nguyen

**By signing above you attest that you have contributed to this submission and confirm that all work you have contributed to this submission is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties.*

Table Of Content

Project Proposal

Project: Smart Greenify Device

Description of the latest similar system:

Similar systems in the market come with specifications like auto watering and grow lights and small pods for growing the plants. Initially, they provide pods or capsules. Inside pods, consumers buy seed from the market or buy it alongside the product, buy soil from the market and use the soil and the seeds to grow plants or herbs according to their requirements. This kind of system has auto watering but the amount of water is fixed. This system also contains grow lights and their timings are fixed.

One of the products that are available inside the market is The Smart Garden 9. The specifications are as listed below

- Automatic watering
- Perfect amount of light thanks to pro-grow lights
- Perfect amount of nutrients and oxygen at the root level
- The companion app to become a plant expert
- Choose from 50+ pre-seeded plant pods from our selection or use your seeds
- A Complimentary set of plant capsules: 3x tomato, 3x basil, 3x lettuce

This system comes with an option of capsules that already have seeds embedded inside them.

Limitations of the latest similar system:

The above-given system is good based on options it gives with the seed capsules so the consumer does not have to buy aftermarket seeds and soil but the limitation of this system is as given below

- The system is only directed towards one type of plants

Different types of plants need different types of atmosphere to grow. When it comes to soil moisture, light intensity & temperature the requirements vary not only in herbs but also in plants. For example plants like Black Olives, Ficus & Areca need high moisture while ornamental plants like Aloe, Jade plant & Cactus need dry soil to grow.

- Does not provide manual control to the consumer

As mentioned in this project all the systems are kept under the automatic mode, this makes the consumer lose control over the plants which the consumer might not like. For Consumers, the privilege of control is still required as it is their plants and they must need an option where they can switch to the manual mode of operation and give commands to the system accordingly. Furthermore, it acts as a backup in case of any failure in the system.

- Does not have sensors to monitor real-time parameters

The project renders the fixed amount of water and light to plants. Therefore there are no sensors to detect whether the amount is sufficient for that particular plant or not. This makes real-time operations hard and makes the project suitable for one type of plant only. Sensors like Soil Moisture, Temperature sensors & Light Sensors, etc could be used in the project to maintain a steady atmosphere for a plant to grow.

- Does not provide real-time data of temperature, moisture, etc.

The growth of plants is a dynamic process therefore the parameters like temperature, soil moisture, etc require continuous. The Ultra-Flexible Continuous checking. This project shows no real-time data of these parameters which is a huge backlash.

- Does not provide remote conductivity

The project did not showcase any kind of remote control technology where a person can monitor, access, and change the system parameters. It could be a useful tool as through cloud storage and IoT we can manage the system remotely.

- Does not provide disease detection

The last and foremost limitation of the project is its inability to detect any kind of disease. As research shows that the untreated disease may cause the death of a plant. Therefore, the detection of the disease is paramount. Even though under perfect conditions, common diseases like Botrytis, Leaf spot, Black Leg can cause serious damage to the plants.

PART I: Initial Solution

(Block Diagram, Features, Hardware and Software Requirement, References):

Concept Introduction

A Study Case:

More and more people are having a hobby of growing plants, herbs indoors, etc. What is the reason behind this trend? In this section, we will be exploring the benefits of interior plants.

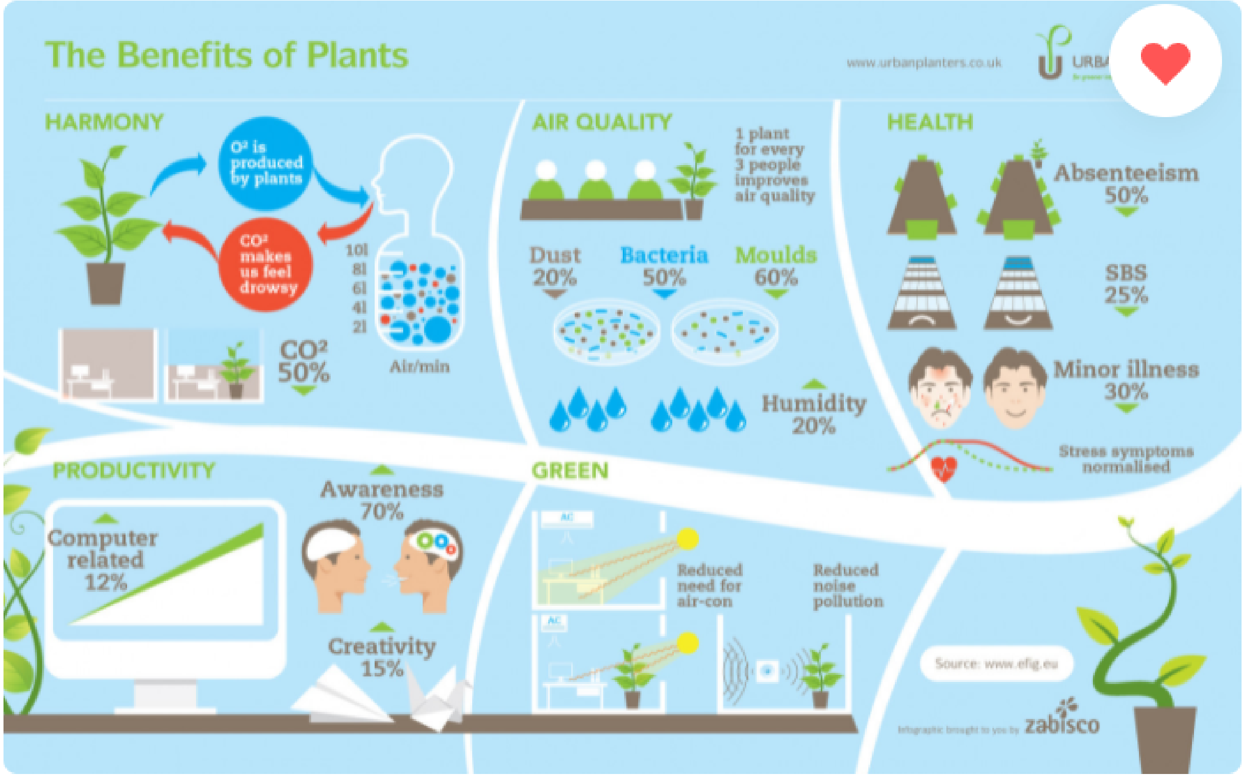
Benefits of growing plants indoors:

- Air: Plants contribute to a cleaner, healthier air for us, hence giving us a better living environment, according to NASA research (NASA, n.d.)
- Humidity: One of the studies (Lohr, 2010, 2) states that foliage plants can increase the relative humidity to healthier and more comfortable levels in interior spaces
- Dust: Adding plants into the room will reduce dust by at least 20% (Lohr, 2010, 2). A study shows that when the plants were in self-watering containers that watered the plants from below, the growing medium surface was dehydrated and dusty. Documenting that interior plants were associated with reduced dust under such circumstances was especially important because it allayed that the ever-increasing medium in containers might make interiors dustier.

More importantly, studies showing that keeping plants indoors also provides some psychological benefits:

- Stress: Interior plants have been associated with reduced stress, increased pain tolerance, and improved people’s productivity. A study showing that when people are in a room with a few containerized interior plants, even when their attention is not drawn to the plants
- Feeling: In a study of people working on computer tasks, there were significant differences in the item "I feel attentive or concentrating" (Lohr et al., 1996). When foliage plants were in the room, people reported feeling more attentive than did people in the room without plants.
- Productivity: Productivity is higher when plants are present—the computer task study mentioned above. People responded significantly more quickly when plants were in the room

than when the plants were absent, and there was no increase in error rate associated with the faster response (Lohr, 2010, 3).



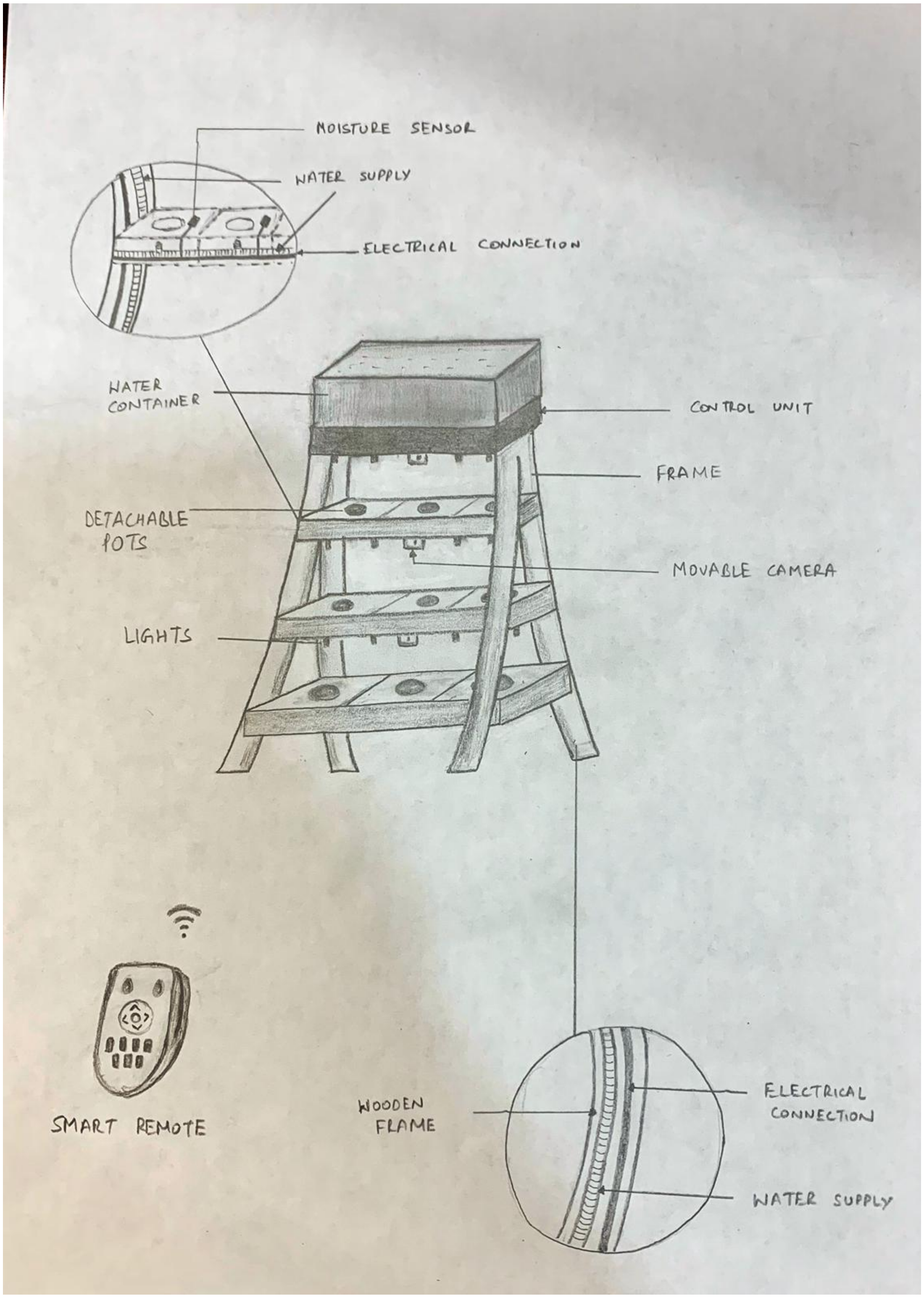
However, these plants need attention and maintenance as well. They need to be monitored and taken care of closely, which requires the amount of time from the owner and knowledge about plants. On the other hand, people are busy with their activities: working, studying, relaxing, etc. therefore, this is one reason people may lose interest in taking care of their little garden. Another reason is: they get bored; many people give up their passion for their hobbies for various reasons (Heshmat, 2017, 1). People like to enjoy their hobbies, especially when they do not have to do much work to care for them. Some new plant owners don't know how to care for a specific plant to keep them alive.

Project Concept: Smart Greenify is a device that helps indoor plants to survive without much effort from the owner.

The device has sensors to detect the water level, humidity, and temperature, and all these parameters can be controlled by the particular plant's requirement. Grow lights will be used to give artificial sunlight to the plants. We will have a log of data of the different varieties of plant species and their requirements stored in the memory. All the data and controls can be accessed on a handheld device wirelessly over Bluetooth. We also will have a camera to monitor the Plant's health.

Image Processing is embedded in the main system since the camera will pick up any weird spots and color changes on the leaves of the plants, then analyze them with a library then notify the user.

Sketch:

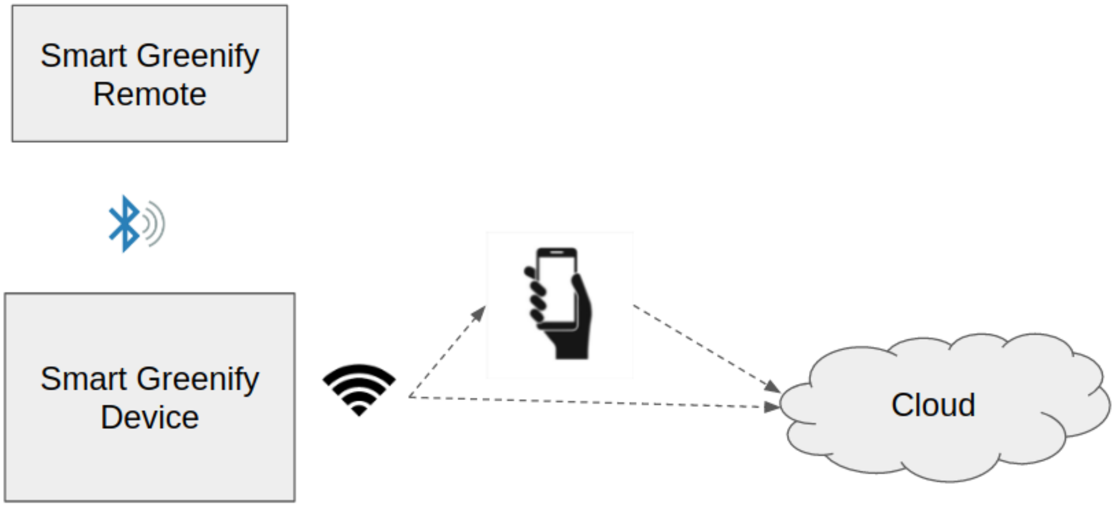

MOISTURE SENSOR
WATER SUPPLY
ELECTRICAL CONNECTION
WATER CONTAINER
CONTROL UNIT
FRAME
DETACHABLE POTS
MOVABLE CAMERA
LIGHTS
SMART REMOTE
WOODEN FRAME
ELECTRICAL CONNECTION
WATER SUPPLY

System-level Description - Initial Block diagram:

Using Wifi and BLE, the real-time information is collected and sent to the owner's smartphone and the data is stored in the cloud via AWS.

Smart Greenify Remote is a small device consisting of a BLE module and some buttons to control the main device via BLE.

Block diagram:



Feature

Key features:

- Auto-adjusting temperature: with an optional enclosed space, the temperature in each slot of the device can be adjusted to a suitable level for any particular plant.
- Auto-adjusting humidity: some plants may enjoy their habitat in a humid environment; our device can replicate the moist air using the humidifier system.
- Auto-watering: the plants will never be dried out again with this feature; the device will detect the soil's humidity and automatically water your plants whenever they need.

- Auto-disease-detect: the device will have a camera system that scans the leaves of your plants, then detects any disease happening to the plants, and recommends proper treatment.
- Light system: with the LEDs system, the device can save up to 70% of power consumption.
- Music player: yes, our device will also have a music function. You wonder why? According to a study, music increases plants' productivity, improves their health and "mood." (Mazlan, 2020).
- Friendly control interface: With a touch screen built on, you can easily control your whole garden within a few steps.
- Smartphone control: the device can be monitored and controlled via the BLE system or by smartphone via Wifi and AWS.

Potential of the project:

Connecting with Alexa, Google Home, Siri to give us updates on our garden whenever we want.

Industrialization: This product can be developed further and apply to a greenhouse farming system, like a mushroom farm that automatically functions.

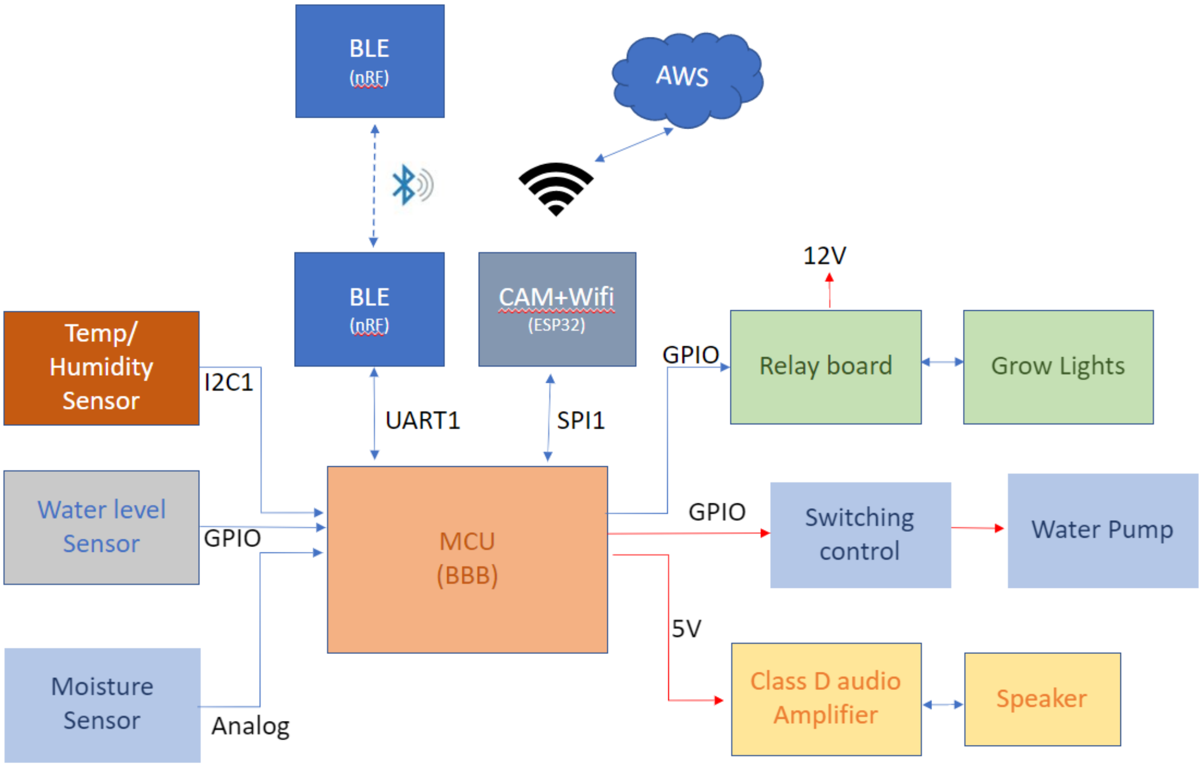
Hardware requirements:

Description:

When it comes to the hardware of this product, the main control unit is the BeagleBone Black. Three kinds of sensors used in this are temperature/humidity sensor, water level sensor, and moisture sensor. So, we detect the moisture of soil, and if it's less, water is pumped from a container below the soil. We also have a water level sensor to fill this water container in case it's too low.

BLE and Wifi modules are present for real-time performance. Grow lights are present so that they can substitute for sunlight. Water supply can be provided by the water pump and a speaker is present to play music as music enhances the growth of plants. A camera is present to get the images of the plants so that disease conditions can be easily found out.

Hardware block diagram



In this block Diagram, Temp/Humidity Sensor, Water level Sensor, and Moisture Sensor are Input modules connected to the MCU through I2C communication, BLE(Bluetooth module) is connected using UART communication and communication with camera and Wifi is using SPI2 communication. MCU will process the readings from Temp/Humidity Sensor, Water level Sensor, and Moisture Sensor and control the Water pump and Grow lights according to the preset parameters.

BLE has two-way communication. So, it will get the reading of moisture, water level, temp, and get the override control of the water pump, Grow lights, and speaker. ESP32 has both camera and Bluetooth connectivity. The input of the camera will be used for image processing. Wi-Fi connectivity will enable mobile connectivity to remote devices, control the water pump, and grow lights to a mobile user.

Components Selection:

Part no.	Qty	Description	Cost	Link
1150	1	Peristaltic Liquid Pum with Silicone Tubing - 12V DC Power	24.95	https://www.adafruit.com/product/1150
997	1	Plastic Water Solenoid Valve - 12V - 1/2" Nominal	6.95	https://www.adafruit.com/product/997
1528-4545-NI	1-2	TUBING PVC 8MM ID X 1 METER	2.18	https://www.digikey.ca/en/products/detail/a afruit-industries-llc/4545/11627733?s=N4I CBcoLQBxVAYygMwIYBsDOBTANCAF ZQDa4ArAEwIC6AvvYVWSACwUcgNA
SEN-13763	1	HUMIDITY & TEMPERATURE SENSOR BR	11.56	https://www.digikey.ca/en/products/detail/s arkfun-electronics/SEN-13763/6023505
4026	**1-2**	**Adafruit STEMMA Soil Sensor - I2C Capacitive Moisture Sensor**	**7.50**	**https://www.adafruit.com/product/4026**
SEN-15569	1	Ultrasonic sensor to detect Water level	5.74	https://www.digikey.ca/en/products/detail/s arkfun-electronics/SEN-15569/10384560
4612	1	Ultra-Flexible White LED Strip - 480 LED per meter - 1m long - Cool White ~6000K	11.95	https://www.adafruit.com/product/4612
4296	**1**	**Raspberry Pi 4 Mod B - 4 GB RAM**	**55.00**	**https://www.adafruit.com/product/4296**
1669	1	Stereo Enclosed Speaker Set - 3W 4 Ohm	7.50	https://www.adafruit.com/product/1669

798	1	Compact Switching Power Supply - Selectable Output 3-12VDC	14.95	https://www.adafruit.com/product/798
14681	1	Electronics Toolkit	79.99	https://elmwoodelectronics.ca/products/146 1
12614	1	Large Breadboard	30.99	https://elmwoodelectronics.ca/products/126 4
		TOTAL COST:	~259.26	

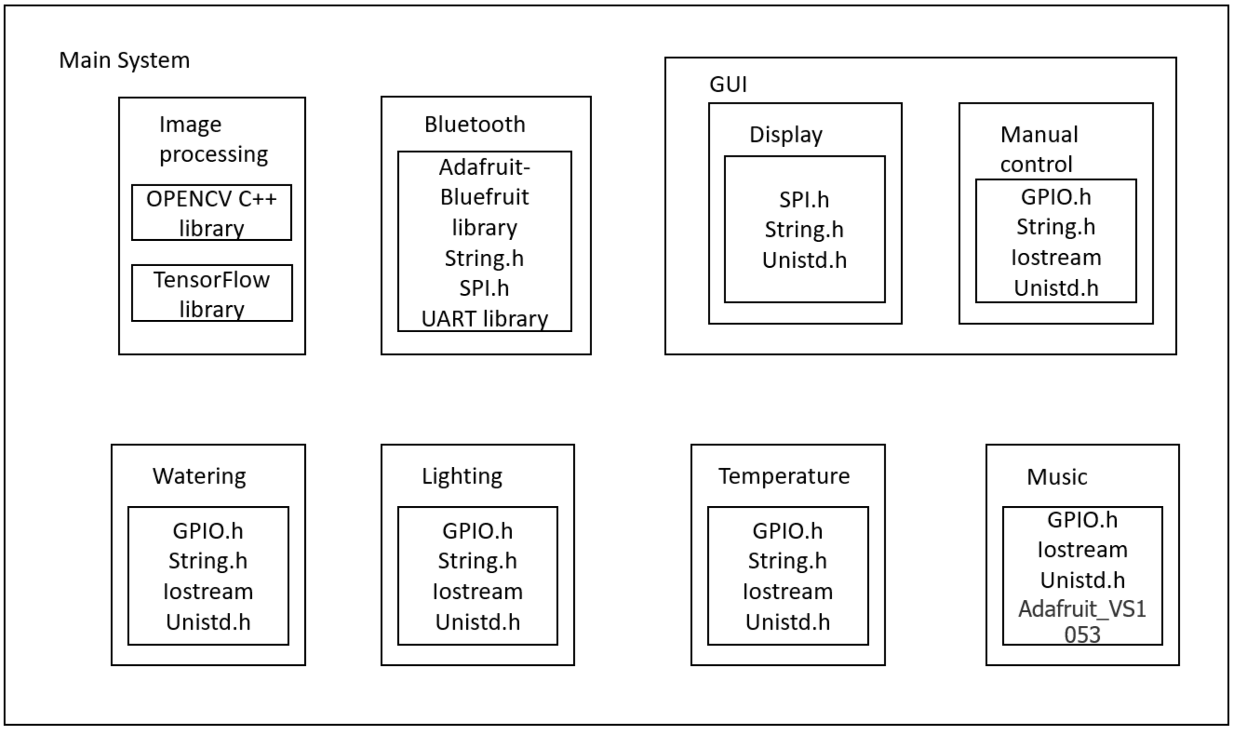
Software requirement:

The device will run on a Debian GNU OS. All features of the devices are written in C++ using the Eclipse IDE and GNU ARM embedded toolchain. AWS is also another important element to create an IoT product.

Design tools that are needed to produce a PCB or simulation circuit in this project included: KiCad and Proteus.

Libraries and API needed for this project can be described in the following architecture.

Software architecture:



Software is divided into eight different functions. The first function is image processing, in this block camera-input would be used to compare to a database using Opencv(c++) to detect plant diseases and also allows sending images to a remote device such as a smartphone connected through Wifi.

The second function is the Bluetooth function. This function will use the Adafruit Bluetooth module and it will use input from remote devices and its output will be to control watering and lightning.

The third and fourth blocks come under API(application programming interface). The display function will send the processed images to the smartphone through wifi connectivity and the manual control function will use the inputs from the smartphone to control watering, lighting, and speaker as output.

The watering function will use the inputs of BLE and ESP42 and sensors(Temp/Humidity sensor, Water level Sensor and Moisture Sensor) and its output will control the water pump.

The Lightning function will use the inputs of the temp sensor, BLE, and ESP32 and its output will control the Grow lights accordingly. The temperature function will use the Temp/Humidity sensor input and will control the lights as output. The music function will control the speaker based on ultra-Flexible. inputs from BLE and ESP32.

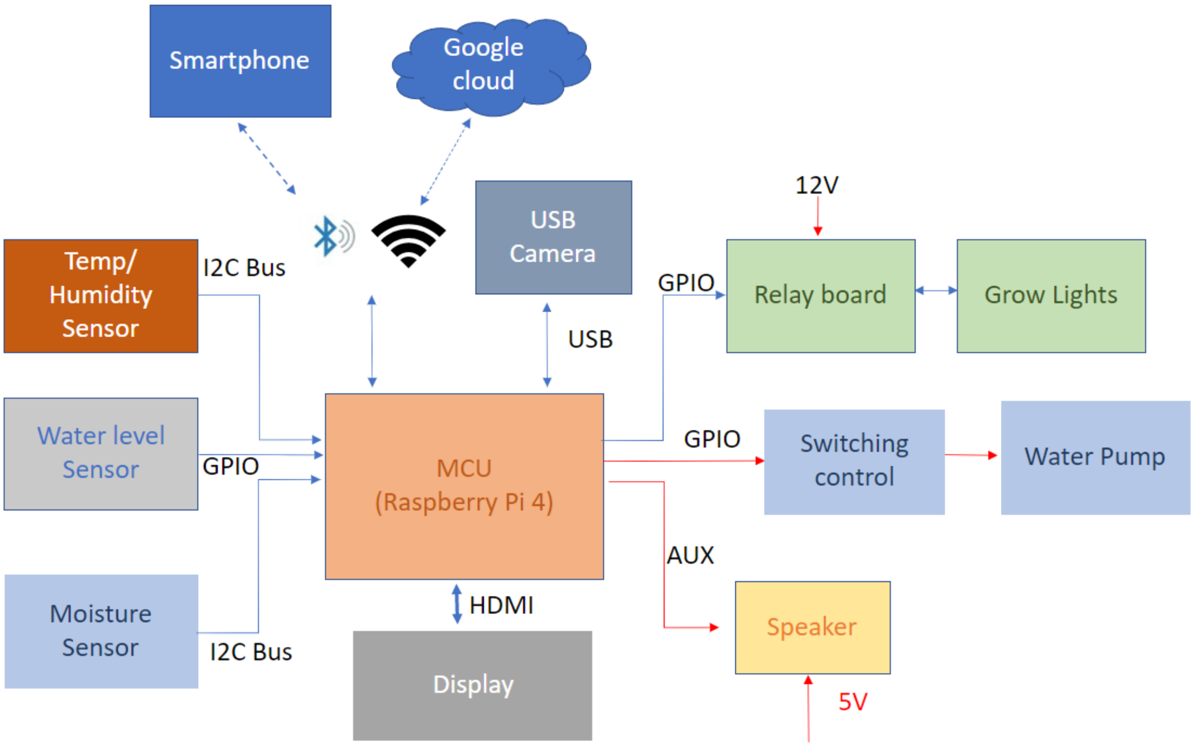
PART II: Final Solution:

Real-Time Operation:

Smart Greenify devices will use the real-time data from sensors and tasks will be created based on the real-time data from the sensors. These tasks will be programmed using pthreads so that they can be completed simultaneously according to the priority of the task. The Raspberry pi 4 MCU that we are using allows us to use pthreads to accommodate real-time operations.

Hardware Requirement

Hardware block diagram



Communication protocols:-

1. Temp/Humidity and Moisture sensors are connected through I2C communication. Pin 3 and 5 on raspberry pi are SDA and SCL With I2C, data is transferred in messages. Messages are broken up

into frames of data. Each message has an address frame that contains the binary address of the slave, and one or more data frames that contain the data being transmitted. The message also includes start and stop conditions, read/write bits, and ACK/NACK bits between each data frame:

- **Start Condition:** The SDA line switches from a high voltage level to a low voltage level *before* the SCL line switches from high to low.
- **Stop Condition:** The SDA line switches from a low voltage level to a high voltage level *after* the SCL line switches from low to high.
- **Address Frame:** A 7 or 10 bit sequence unique to each slave that identifies the slave when the master wants to talk to it.
- **Read/Write Bit:** A single bit specifying whether the master is sending data to the slave (low voltage level) or requesting data from it (high voltage level).
- **ACK/NACK Bit:** Each frame in a message is followed by an acknowledge/no-acknowledge bit. If an address frame or data frame was successfully received, an ACK bit is returned to the sender from the receiving device.

2. Water level sensor, Relay board, switching control is connected to MCU through GPIOs. GPIOs can be configured as Input or Output on the basis of the programmer's requirement. The operating voltage of the GPIO pins is 3.3v with a maximum current draw of 16mA. The GPIO available on Raspberry pi is only digital, it means it can either be zero or one and the direction (output or input) can be configured by the programmer.

3. Display is connected through the HDMI protocol. Raspberry pi 4 has 2 micro HDMI slots available onboard. Standard or Mini HDMI ports can be converted to Micro HDMI ports using converter wires. HDMI uses transition minimized differential signaling (TMDS) to move information from one place to another. TMDS is a way of encoding the signal to protect it from degrading as it travels down the length of the cable. Here's what happens:

- The device which is used to send the signal like an HD-DVD player, encodes the signal and to decrease the number of transitions between ON and OFF. The steps involved in encoding helps to protect the quality of signal by reducing the number of chances for the signal to degrade.

- The signal is carried by the cables in the twisted pair and the pair carries a copy in the inverse of the signal.
- The loss of any signal is calculated by the difference or differential between its inverse and the signal. Thus this information is used for compensation which is carried through the decoding by the receiving device just like a HDTV.

4. USB protocol is used to connect the camera to MCU. Raspberry pi has 4 USB slots available onboard 2 of them are USB 2.0 and two are USB3.0. When the power comes to the host it assigns each bus which is connected to its corresponding device an address. This process is also called Enumeration as all the devices are also enumerated when they connect to the bus.

After that the main function of the host is to find out what type of transfer of data a specific device will perform. There are mainly three types of data transfer - Bulk, Isochronous & Interrupt. In our case, the data flows between the device and the host in real time and also there is no error correction)

The host can also send commands or query parameters with control packets. As devices are enumerated, the host is keeping track of the total bandwidth that all of the isochronous and interrupt devices are requesting. They can consume up to 90 percent of the 480 Mbps of bandwidth that's available (USB 3.0 increases that speed to 4.8 gigabits per second). After 90 percent is used up, the host denies access to any other isochronous or interrupt devices. Control packets and packets for bulk transfers use any bandwidth leftover (at least 10 percent).

The Universal Serial Bus divides the available bandwidth into frames, and the host controls the frames. Frames contain 1,500 bytes, and a new frame starts every millisecond. During a frame, isochronous devices get a slot so they're guaranteed the bandwidth they need. Bulk and control transfers use whatever space is left.

5. Speaker is connected to MCU through the inbuilt auxiliary(Aux) port on Raspberry pi 4. The aux port is a 4-pole Audio jack This style of connector is sometimes referred to as "TRRS", which stands for "Tip-Ring-Ring-Sleeve". The four conductors carry video, left audio, right audio, and ground. Cables are easy to obtain but use different configurations.

6. Raspberry pi 4 has an inbuilt BLE(Bluetooth low energy) and a Wifi combo chipset integrated onboard. Raspberry pi which can be configured to be connected to a smartphone. Bluetooth Low Energy (BLE) is a subset of Bluetooth Classic and was first introduced as one of the core specs of Bluetooth 4.0 in 2010.BLE uses a 2.4GHz frequency and has 40 channels (2402-2480MHz). When coming to the device via BLE, the device will work as a peripheral with a custom UUID (Universally unique identifier) that acts like an address where it can be accessed on a Smartphone (which will work as a central).

7. Wifi

The easiest way to connect your Raspberry Pi to a wireless network is to use the desktop tool. However, this means that you will need to set it up with a keyboard, mouse, and display. The alternative is to first hook up an Ethernet cable, then connect via VNC(Virtual Network Computing) or RDP(Remote Desktop Protocol). Just remember to disconnect Ethernet when the Pi is connected wirelessly. For more information following this article How to Set Up Wi-Fi and Bluetooth on Raspberry Pi

Hardware embedded tools:-

1.Laptop

2.Soldering Iron

3.Raspberry pi 4 will the main control unit

BOM:

Part no.	Qty	Description	Cost	Link
1150	1	Peristaltic Liquid Pum with Silicone Tubing - 12V DC Power	24.95	https://www.adafruit.com/product/1150
997	1	Plastic Water Solenoid Valve - 12V - 1/2" Nominal	6.95	https://www.adafruit.com/product/997
1528-4545-NI	1-2	TUBING PVC 8MM ID X 1 METER	2.18	https://www.digikey.ca/en/products/detail/aafruit-industries-llc/4545/11627733?s=N4IgCBcoLQBxVAYygMwIYBsDOBTANCAPQDa4ArAEwIC6AvvYVWSACwUcgNA
SEN-13763	1	HUMIDITY & TEMPERATURE SENSOR BR	11.56	https://www.digikey.ca/en/products/detail/sarkfun-electronics/SEN-13763/6023505
4026	**1-2**	**Adafruit STEMMA Soil Sensor - I2C Capacitive Moisture Sensor**	**7.50**	**https://www.adafruit.com/product/4026**
SEN-15569	1	Ultrasonic sensor to detect Water level	5.74	https://www.digikey.ca/en/products/detail/sarkfun-electronics/SEN-15569/10384560
4612	1	Ultra-Flexible White LED Strip - 480 LED per meter - 1m long - - Cool White ~6000K	11.95	https://www.adafruit.com/product/4612
4296	**1**	**Raspberry Pi 4 Mode B - 4 GB RAM**	**55.00**	**https://www.adafruit.com/product/4296**
4884	1	Adafruit Feather RP2040	11.95	https://www.adafruit.com/product/4884
1669	1	Stereo Enclosed	7.50	https://www.adafruit.com/product/1669

		Speaker Set - 3W 4 Ohm		
798	1	Compact Switching Power Supply - Selectable Output 3-12VDC	14.95	https://www.adafruit.com/product/798
181	1	Standard LCD 16x2 + extras - white on blue	9.95	https://www.adafruit.com/product/181
292	1	i2c / SPI character LCD backpack	9.95	https://www.adafruit.com/product/292
		TOTAL COST:	212.0	

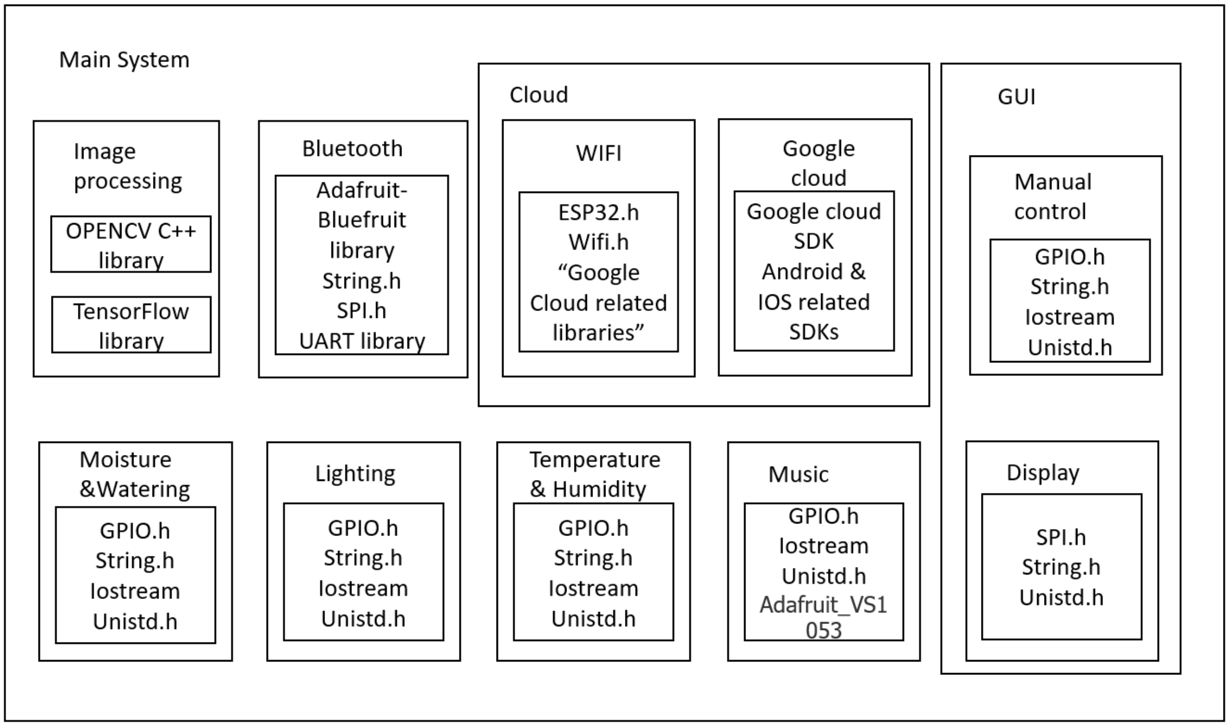
Features:

- Smartphone control: the device can be monitored and controlled via the BLE system or by smartphone via Wifi and Google Cloud.
- Auto-adjusting temperature: with an optional enclosed space, the temperature in each slot of the device can be adjusted to a suitable level for any particular plant.
- Auto-adjusting humidity: some plants may enjoy their habitat in a humid environment; our device can replicate the humidifier system's moist air humidifier system's moist air-control.
- Auto-watering: the plants will never be dried out again with this feature; the device will detect the soil's humidity and automatically water your plants whenever they need.
- Auto-disease-detect: the device will have a camera system that scans the leaves of your plants, then detects any disease happening to the plants, and recommends proper treatment.
- Light system: with the LEDs system, the device can save up to 70% of power consumption.
- Music player: yes, our device will also have a music function. You wonder why? According to a study, music increases plants' productivity, and it improves their health and "mood." (Mazlan, 2020).

- Friendly control interface: With a touch screen built on, you can easily control your whole garden within a few steps.

Software Requirement:

Software Architecture



The Cloud block is added to this architecture. Cloud service will require a wifi module connected from the main system (Raspberry PI 4) and Google Cloud for storage and end-user app on mobile.

Software embedded Tools:

1. Kicad and proteus for design purpose
2. Eclipse IDE for C/C++ programing
3. Raspbian for Raspberry pi

Flowchart:

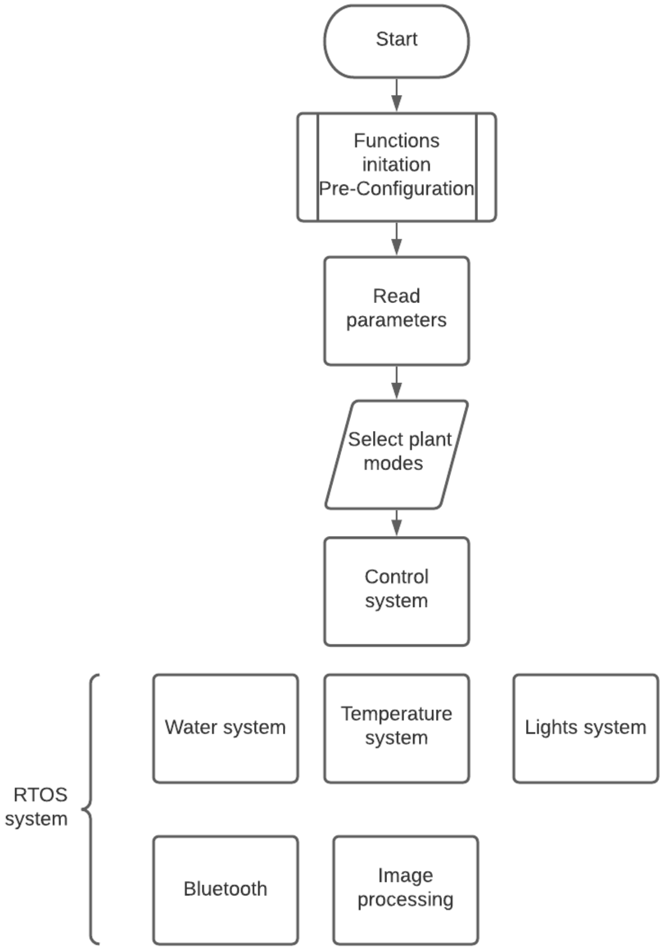
Description:

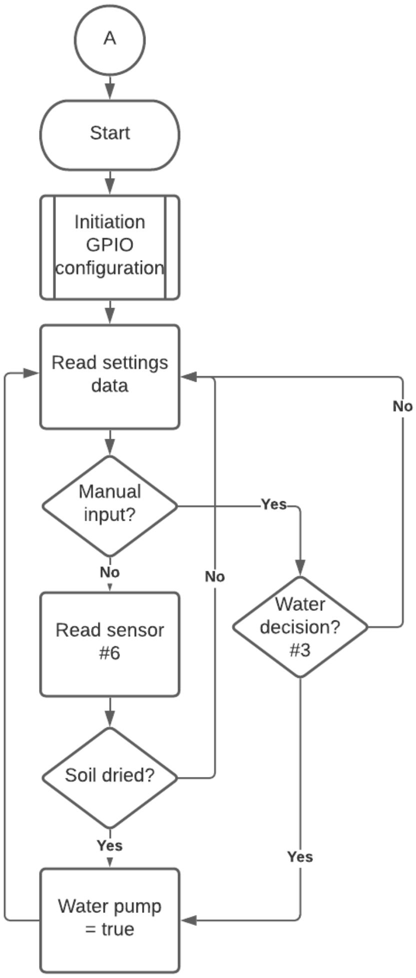
The flowcharts below include nine parts: One main function and eight sub-functions

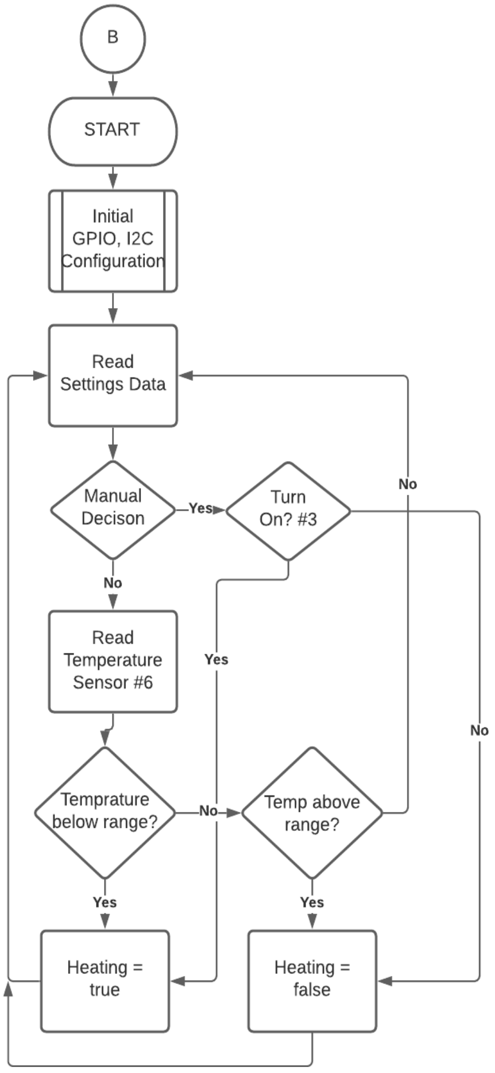
The main function has the responsibility to monitor and control other functions. The sub-functions are:

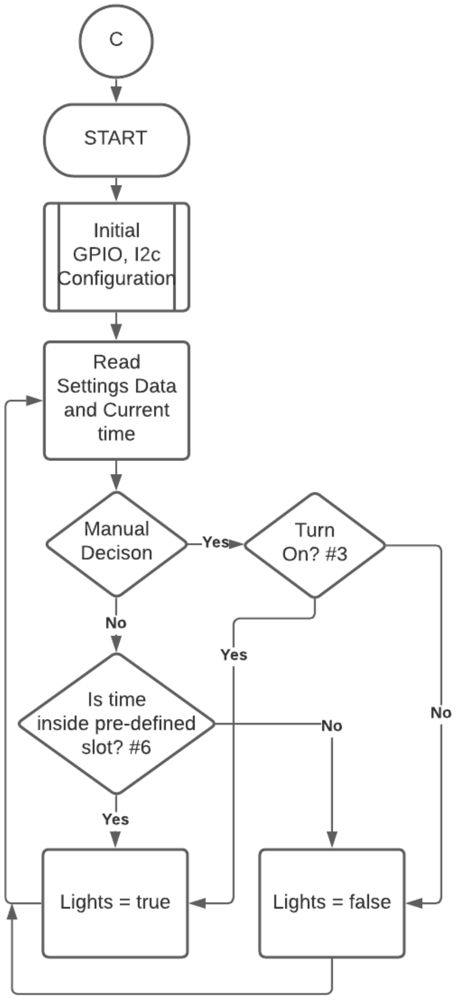
- A for Water system
- B for Temperature system.
- C for Light system.
- D for BLE control.
- E for Cloud control.
- F for Disease detection system.
- G & H for GUI - Control and Display.
- I for Music system

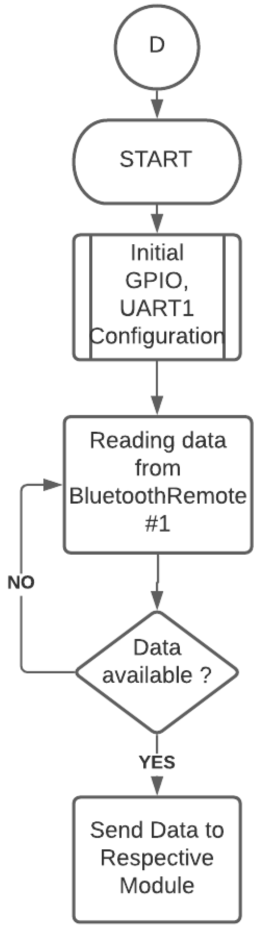
Sign # followed by a number inside the flowchart defines the priority of the task

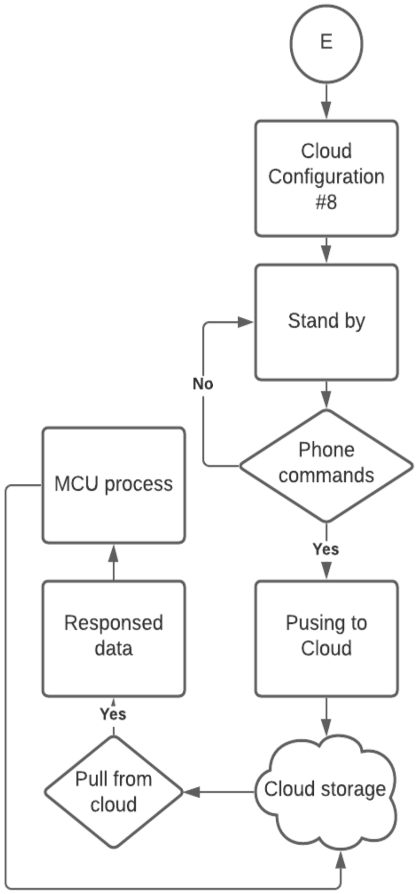


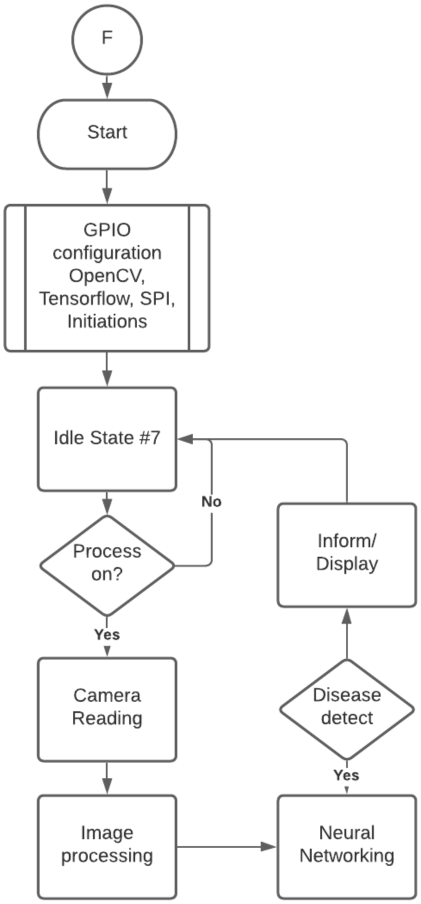


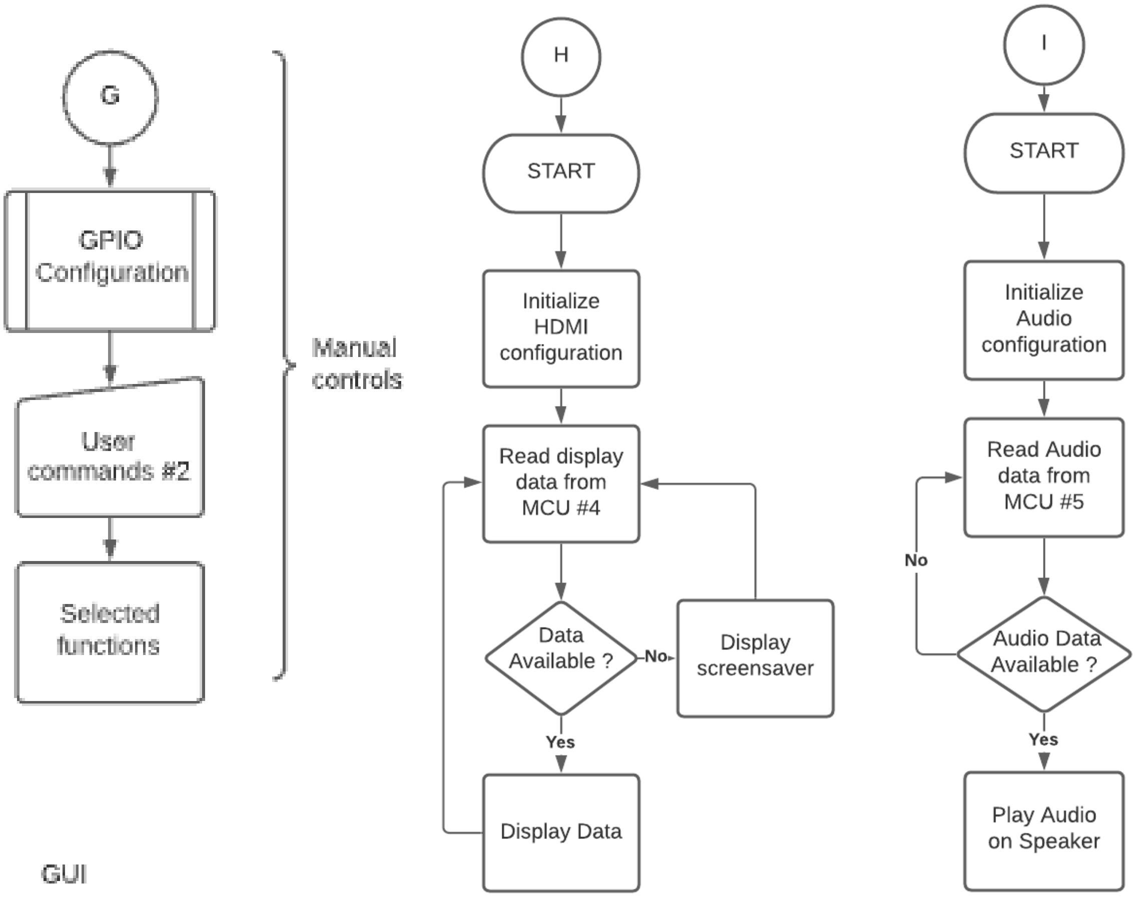

B
START
Initial GPIO, I2C Configuration
Read Settings Data
Manual Decison
Yes
Turn On? #3
No
Read Temperature Sensor #6
Yes
No
No
Temprature below range?
No
Temp above range?
Yes
Yes
Heating = true
Heating = false


C
START
Initial GPIO, I2c Configuration
Read Settings Data and Current time
Manual Decison
Yes
Turn On? #3
No
Yes
Is time inside pre-defined slot? #6
No
No
Yes
Lights = true
Lights = false


D
START
Initial GPIO, UART1 Configuration
Reading data from BluetoothRemote #1
NO
Data available ?
YES
Send Data to Respective Module


E
Cloud Configuration #8
Stand by
No
Phone commands
Yes
Pusing to Cloud
Cloud storage
Pull from cloud
Yes
Responsed data
MCU process


F
Start
GPIO configuration OpenCV, Tensorflow, SPI, Initiations
Idle State #7
Process on?
No
Yes
Camera Reading
Image processing
Neural Networking
Yes
Disease detect
Inform/ Display



Programing language and standard selection

The latest standard of Embedded C language

BARR-C:2018, the stylistic coding rules have been fully harmonized with MISRA C: 2012, while helping embedded system designers reduce defects in firmware written in C and C++. For more information, we can refer to the Embedded C Coding Standard.

In this standard, we use some of the rules for coding like:

- general rules;
- comments rules;
- white spaces rules;

- module rules;
- datatype rules;
- procedure rules;
- variable rules;
- statement rules

For instance:

Booleans

Rules:

- *Boolean variables shall be declared as type bool.*
- *Non-Boolean values shall be converted to Boolean via the use of relational operators (e.g., < or !=), not via casts.*

Example:

```
#include <stdbool.h>
...
 bool b_in_motion = (0 != speed_in_mph);
```

Reasoning: The C90 standard did not define a data type for Boolean variables and C programmers have widely treated any non-zero integer value as true. The C99 language standard is backward compatible with this old-style, but also introduced a new data type for Boolean variables along with new constants true and false in the stdbool.h header file.

Initialization

Rules:

- *All variables shall be initialized before use.*
- *It is preferable to define local variables as you need them, rather than all at the top of a function.*
- *If project- or file-global variables are used, their definitions shall be grouped together and placed at the top of a source code file.*
- *Any pointer variable lacking an initial address shall be initialized to NULL.*

Example:

```
uint32_t g_array[NUM_ROWS][NUM_COLS] = { ... };
...
```

```
for (int col = 0; col < NUM_COLS; col++)
   {
       g_array[row][col] = ...;
   }
```

Reasoning: Too many programmers assume the C run-time will watch out for them, e.g., by zeroing the value of uninitialized variables on system startup. This is a bad assumption, which can prove dangerous in a mission-critical system. For readability reasons it is better to declare local variables as close as possible to their first use, 10 which C99 makes possible by incorporating that earlier feature of C++.

The latest standard of C++ language

C++17 is the latest version of C++ available. C++ is a general-purpose programming language based on the C programming language as described in ISO/IEC 9899:2011 Programming languages -C. In addition to the facilities provided by C, C++ provides additional data types, classes, templates, exceptions, namespaces, operator overloading, function name overloading, references, free store management operators, and additional library facilities.

For more information following the C++ 17 document

Some instances:

Declaration:

Declarations generally specify how names are to be interpreted. Declarations have the form

declaration-seq:

declaration

declaration-seq declaration

declaration:

block-declaration

nodeclspec-function-declaration

function-definition

template-declaration

deduction-guide

explicit-instantiation

explicit-specialization

linkage-specification

namespace-definition

empty-declaration

Attribute-declaration

block-declaration:

simple-declaration

asm-definition

namespace-alias-definition

using-declaration

using-directive

static_assert-declaration

alias-declaration

opaque-enum-declaration

Nodeclspec-function-declaration:

attribute-specifier-seq$_{opt}$ *declarator ;*

Alias-declaration:

using identifier attribute-specifier-seq$_{opt}$ *= defining-type-id ;*

Simple-declaration:

decl-specifier-seq init-declarator-list$_{opt}$ *;*

attribute-specifier-seq decl-specifier-seq init-declarator-list ;

attribute-specifier-seq$_{opt}$ *decl-specifier-seq ref-qualifieropt [identifier-list] initializer ;*

Static_assert-declaration:

static_assert (constant-expression) ;

static_assert (constant-expression , string-literal) ;

Empty-declaration:

;

Attribute-declaration:

attribute-specifier-seq ;

<u>*Example:*</u>

In a *simple-declaration*, the optional *init-declarator-list* can be omitted only when declaring a class or enumeration. that is when the *decl-specifier-seq* contains either a class-specifier, an *elaboratedtype-specifier* with a class-key (12.1), or an enum-specifier. In these cases and whenever a class-specifier or enum-specifier is present in the *decl-specifier-seq*, the identifiers in these specifiers are among the names being declared by the declaration (as class-names, enum-names, or enumerators, depending on the syntax). In such cases, the *decl-specifier-seq* shall introduce one or more names into the program or shall redeclare a name introduced by a previous declaration

```
enum { }; // ill-formed
typedef class { }; // ill-formed
```

— end example]

Task table:

Task	Day start	Day End	Person in charge
Project Proposal	May 10	June 3	Group
Testing components	Jun 10	Jun 14	Manu
Setting up Raspberry Pi	Jun 14	Jun 17	Amonjot
Interfacing humidity and temperature sensor	Jun 17	Jun 24	Vy
Interfacing Water level sensor	Jun 17	Jun 24	Anna
Interfacing Moisture sensor	Jun 17	Jun 24	Manu
Interfacing Display	Jun 17	Jun 24	Vy
Interfacing USB camera	Jun 24	Jun 30	Anna
Interfacing Speaker	Jun 24	Jun 30	Amonjot
Interfacing with Image processing	Jun 30	July 14	Anna
Interfacing grow lights and relay board	Jun 30	July 14	Manu
Interfacing water pump and switch control	Jun 30	July 14	Amonjot

Google cloud interfacing	July 14	Aug 5	Anna
Smartphone interfacing	July 14	Aug 5	Vy
Schematic design for the whole system	Aug 5	Aug 12	Vy
PCB layout design	Aug 5	Aug 12	Manu
Integrating & real-time programming the whole system	Aug 12	Aug 15	Amonjot
Report writing	July 29	Aug 15	Group
Demonstration	Aug 16	Aug 19	Group

Environmental Issue

First of all, Smart Greenify Device is an environment-oriented product, our device will help to improve the quality of life in your house.

Benefits of growing plants indoors:

- Air: Plants contribute to a cleaner, healthier air for us, hence giving us a better living environment, according to NASA research (NASA, n.d.)
- Humidity: One of the studies (Lohr, 2010, 2) states that foliage plants can increase the relative humidity to healthier and more comfortable levels in interior spaces
- Dust: Adding plants into the room will reduce dust by at least 20% (Lohr, 2010, 2). A study shows that when the plants were in self-watering containers that watered the plants from below, the growing medium surface was dehydrated and dusty. Documenting that interior plants were associated with reduced dust under such circumstances was especially important because it allayed that the ever-increasing medium in containers might make interiors dustier.

More importantly, studies showing that keeping plants indoors also provides some psychological benefits:

- Stress: Interior plants have been associated with reduced stress, increased pain tolerance, and improved people's productivity. A study showing that when people are in a room with a few containerized interior plants, even when their attention is not drawn to the plants

- Feeling: In a study of people working on computer tasks, there were significant differences in the item "I feel attentive or concentrating" (Lohr et al., 1996). When foliage plants were in the room, people reported feeling more attentive than did people in the room without plants.
- Productivity: Productivity is higher when plants are present—the computer task study mentioned above. People responded significantly more quickly when plants were in the room than when the plants were absent, and there was no increase in error rate associated with the faster response (Lohr, 2010, 3).

That is why our team has the vision to create Smart Greenify Device.

On the other hand, electronic waste is one of the issues that our team has considered during the process of creating this design. Ifixit.com state that:

“When electronics end up in landfills, toxins like lead, mercury, and cadmium leach into the soil and water.

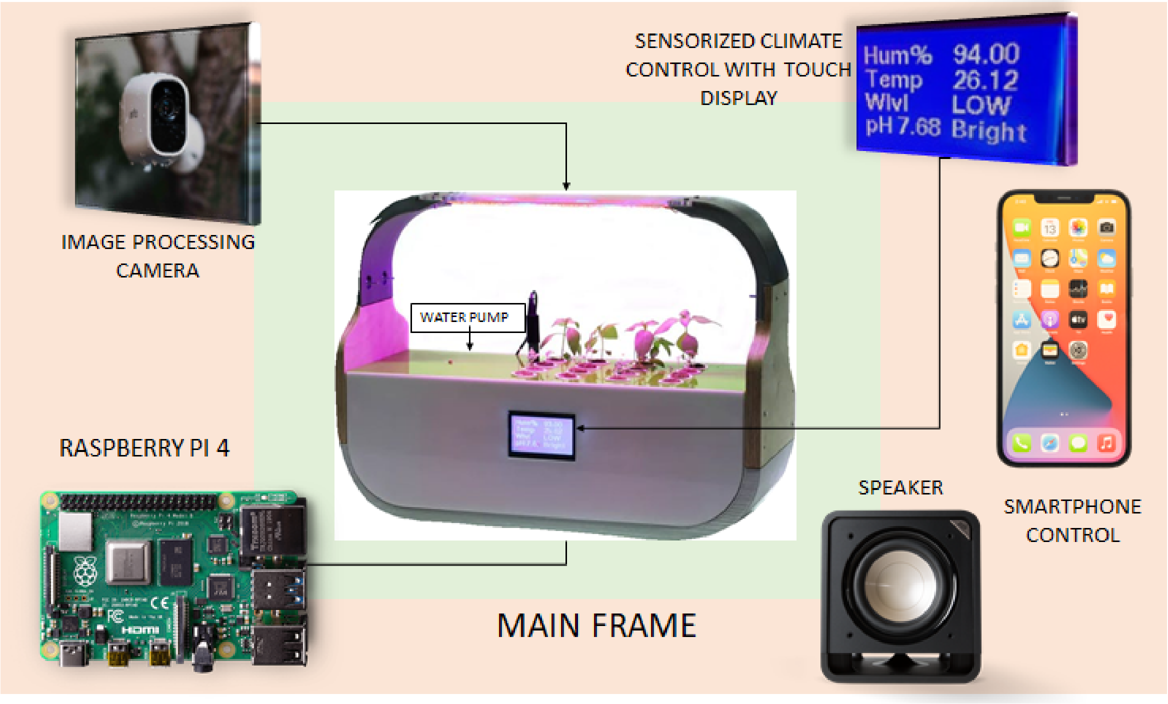
The electronic waste problem is huge: More than 48 million tons of e-waste are produced every year. If you put every blue whale alive today on one side of a scale and one year of US e-waste (6.9 million tons) on the other, the e-waste would be heavier.”

Also according to Ifixit.com, the matter has even become bigger when the E-waste from developed countries will be shipped to poorer countries for TEQs (Tradable Energy Quotas), but the children living nearby these junkyards will burn this waste for craps though they do not understand how toxic and harmful it is.

Global consumption of electronics is increasing. Every year we create more e-waste than before. At least 50% of Africa’s e-waste comes from within the continent. China discards 750 million electronic devices a year.

By considering those reasons, our team has come up with an idea that our product can be recollected after use and recycle all the components. In this way, we will reduce less electronic waste to the environment.

Engineering Design



References

1.Heshmat, S. (2017, 06 16). Eight Reasons Why We Get Bored. Psychology Today. https://www.psychologytoday.com/us/blog/science-choice/201706/eight-reasons-why-we-get-bored

2.Lohr, V. L. (2010). What Are the Benefits of Plants Indoors and Why Do We Respond Positively to Them? Department of Horticulture and Landscape Architecture. https://public.wsu.edu/~lohr/pub/2010LohrBenefitsPltsIndoors.pdf

3.Mazlan. (2020, 03 20). The Effect of Music on Plant Growth. Dengarden. Retrieved 02 26, 2021, from https://dengarden.com/gardening/the-effect-of-music-on-plant-growth#:~:text=According%20to%20some%20studies%2C%20jazz,country%2C%20jazz%2C%20or%20classical

4.NASA. (n.d.). Plants Clean Air and Water for Indoor Environments. Plants Clean Air and Water for Indoor Environments. Retrieved 02 26, 2021, from https://spinoff.nasa.gov/Spinoff2007/ps_3.html

5. AG, I. T. (n.d.). LED Lighting Tech for Urban Farming - Infineon Technologies. Www.infineon.com. https://www.infineon.com/cms/en/discoveries/LEDs-in-Urban-Farming/

6. Plants and Soil Moisture Information | Greener on the Inside. (n.d.). Retrieved June 2, 2021, from https://www.ambius.com/blog/plants-and-soil-moisture/

7. Raspberry Pi 4 Model B specifications – Raspberry Pi. (2019). Raspberrypi.org. https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/

8. BeagleBoard.org - black. (n.d.). Beagleboard.org. https://beagleboard.org/black

9. Preimesberger, C. (2021, March 15). Google Cloud vs AWS | 2021 Comparison by eWeek. EWEEK. https://www.eweek.com/cloud/aws-vs-google-cloud-platform/

10 .Rego. (n.d.). Working with PThreads in Raspberry Pi. Gist. Retrieved June 2, 2021, from https://gist.github.com/pvrego/7fd2610d0bf5ca063e63db6ab776e89c

11. Install pthreads on raspbian - Raspberry Pi Forums. (n.d.). Www.raspberrypi.org. Retrieved June 2, 2021, from https://www.raspberrypi.org/forums/viewtopic.php?t=174510

12. c - Pthread-barriers and raspberry pi. (n.d.). Stack Overflow. Retrieved June 2, 2021, from https://stackoverflow.com/questions/49694164/pthread-barriers-and-raspberry-pi

13. webmaster. (2016, May 26). Embedded C Coding Standard. Barr Group. https://barrgroup.com/Embedded-Systems/Books/Embedded-C-Coding-Standard

14 Bellairs, R. (n.d.). Should I Adopt C++17? Perforce Software. Retrieved June 2, 2021, from https://www.perforce.com/blog/qac/should-i-adopt-cpp17

15 Electronics | 44, C. B. | D. (2016, February 13). Basics of the I2C Communication Protocol. Circuit Basics. https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/

16 C++17 - cppreference.com. (n.d.). En.cppreference.com. Retrieved June 2, 2021, from https://en.cppreference.com/w/cpp/17

17 How HDMI Works. (2007, October 8). HowStuffWorks. https://electronics.howstuffworks.com/hdmi

18 How USB Ports Work. (2000, April 1). HowStuffWorks. https://computer.howstuffworks.com/usb.htm#pt3

19 Raspberry Pi 4-pole Audio/Video Jack. (2014, July 22). Raspberry Pi Spy. https://www.raspberrypi-spy.co.uk/2014/07/raspberry-pi-model-b-3-5mm-audiovideo-jack/

20. How to Set Up Wi-Fi and Bluetooth on Raspberry Pi. (2019, December 16). MUO. https://www.makeuseof.com/tag/setup-wi-fi-bluetooth-raspberry-pi-3/

21. E-waste is the Toxic Legacy of our Digital Age. (n.d.). Retrieved from https://www.ifixit.com/Right-to-Repair/E-waste

Instructor’s Remarks: